Oracle Private Cloud Appliance
Upgrade Guide

F78218-08
March 2024

Oracle Private Cloud Appliance Upgrade Guide,

F78218-08

Contents

7 Upgrading Individual Components

Preface

This publication is part of the customer documentation set for Oracle Private Cloud Appliance Release 3.0. Note that the documentation follows the release numbering scheme of the appliance software, not the hardware on which it is installed. All Oracle Private Cloud Appliance product documentation is available at https://docs.oracle.com/en/engineered-systems/private-cloud-appliance/index.html.

Oracle Private Cloud Appliance Release 3.x is a flexible general purpose Infrastructure as a Service solution, engineered for optimal performance and compatibility with Oracle Cloud Infrastructure. It allows customers to consume the core cloud services from the safety of their own network, behind their own firewall.

Audience

This documentation is intended for owners, administrators and operators of Oracle Private Cloud Appliance. It provides architectural and technical background information about the engineered system components and services, as well as instructions for installation, administration, monitoring and usage.

Oracle Private Cloud Appliance has two strictly separated operating areas, known as enclaves. The Compute Enclave offers a practically identical experience to Oracle Cloud Infrastructure: It allows users to build, configure and manage cloud workloads using compute instances and their associated cloud resources. The Service Enclave is where privileged administrators configure and manage the appliance infrastructure that provides the foundation for the cloud environment. The target audiences of these enclaves are distinct groups of users and administrators. Each enclave also provides its own separate interfaces.

It is assumed that readers have experience with system administration, network and storage configuration, and are familiar with virtualization technologies. Depending on the types of workloads deployed on the system, it is advisable to have a general understanding of container orchestration, and UNIX and Microsoft Windows operating systems.

Feedback

Provide feedback about this documentation at https://www.oracle.com/goto/docfeedback.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.

Convention	Meaning
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, code in examples, text that appears on the screen, or text that you enter.
`$` prompt	The dollar sign (`$`) prompt indicates a command run as a non-root user.
`#` prompt	The pound sign (`#`) prompt indicates a command run as the `root` user.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

Upgrading Your Oracle Private Cloud Appliance

This guide provides instructions for an administrator to upgrade the Oracle Private Cloud Appliance or one of its components.

Do not install or upgrade individual packages on the appliance components. Only upgrades as described in this chapter are supported. Security and other updates are provided through patches. Patching is separate from the upgrade functionality and uses a ULN mirror to download supported packages to the shared storage on the management nodes.

Implementation details and technical background information for the upgrade and patching functionality can be found in the Oracle Private Cloud Appliance Concepts Guide. Refer to the sections "Upgrade" and "Patching" in the chapter Appliance Administration Overview.

Patching instructions are provided in a separate document. Refer to the Oracle Private Cloud Appliance Patching Guide.

Oracle-provided Images
At the end of the full management node cluster upgrade procedure, the Oracle Cloud Infrastructure images are automatically updated. The management node upgrade procedure includes the steps described in Upgrading Oracle Cloud Infrastructure Images so they do not need to be performed separately. However, on systems with software versions 3.0.2-b852928 and earlier, those steps are not automated and must always be performed to obtain the latest images.

Upgrade Strategy
We recommend to run the latest available software on your Oracle Private Cloud Appliance. It improves protection against vulnerabilities and allows you to take advantage of all new features, bug fixes, and functional improvements.
The latest Upgrader code automatically enforces prerequisite software versions. During the upgrade or patch preparations, the Upgrader service validates the currently installed appliance software version against the new target version. If the appliance is not running at least the minimum required version, the Upgrader exits the process and rolls back the environment to its previous state. You must first install the prerequisite version as indicated in the log.

Upgrade Order
Components must be upgraded in a prescribed order. In appliance software version 3.0.2-b892153 and later, the upgrade plan helps manage the order of upgrade operations. When upgrading to version 3.0.2-b1081557 or later, there is an extra requirement to upgrade the ZFS Storage Appliance firmware before all other components. For more information, see Checking Upgrade Plan Status and Progress.

2
Upgrade Requirements

> **Caution:**
>
> The granular appliance architecture with built-in redundancy allows administrators to upgrade or patch components without downtime. However, resource capacity and performance might be reduced while an upgrade or patch workflow is in progress.
>
> We recommend that administrators responsible for upgrade or patching notify all Compute Enclave users in advance about such planned maintenance operations.
>
> This is particularly important for users of the Oracle Container Engine for Kubernetes (OKE), because new cluster deployments are not allowed during the maintenance window, and some types of application clusters might experience service interruptions.

Before you start an upgrade procedure, ensure that you have the required permissions and have downloaded the ISO image to a suitable location.

Verifying Permissions

To be able to execute an upgrade, you must have an administrator account to log in to the Service Enclave. You must be a member of one of these authorization groups: SuperAdmin, Admin, or DR Admin. More information can be found in the chapter Administrator Account Management of the Oracle Private Cloud Appliance Administrator Guide.

When you log in to the Service CLI, you can verify that the upgrade commands are available to you by displaying all custom commands. The list of commands is filtered based on your access profile. If the upgrade commands are listed, it means you have permission to execute them.

```
PCA-ADMIN> showallcustomcmds
    Operation Name: <Related Object(s)>
    ----------------------------------
    [...]
    getUpgradeJob:  UpgradeJob
    getUpgradeJobs:  UpgradeJobList
    getUpgradeRequests:  UpgradeRequest
    killUpgradeJob:  UpgradeJob
    [...]
    upgradeCN:  UpgradeRequest
    upgradeEtcd:  UpgradeRequest
    upgradeFullMN:  UpgradeRequest
    upgradeHost:  UpgradeRequest
    upgradeIlom:  UpgradeRequest
    upgradeKubernetes:  UpgradeRequest
    upgradeMySQL:  UpgradeRequest
    upgradePlatform:  UpgradeRequest
```

```
upgradeSwitch:  UpgradeRequest
upgradeVault:  UpgradeRequest
upgradeZfssa:  UpgradeRequest
```

Preparing the Upgrade Environment

Software versions and upgrades for Oracle Private Cloud Appliance are made available for download through My Oracle Support. The ISO file contains all the files and packages required to upgrade the appliance hardware and software components to a given release. All the items within the ISO file have been tested to work with each other and qualified for installation on your rack system.

To be able to use an ISO file to upgrade your appliance, you need to download the file to a location from where a web server can make it available to the Private Cloud Appliance management nodes. If you have set up a bastion host connected to the internal administration network of the appliance, it is convenient to store the ISO file on that machine and run a web server to make the ISO file accessible over http.

Before performing any upgrade operations, you unpack the contents of the ISO file to populate the source directories in the shared storage that is mounted on all three management nodes. This ensures that the new version is installed when an upgrade command is executed.

Prerequisite Version
The latest Upgrader code automatically enforces prerequisite software versions on your Private Cloud Appliance. In the early stages of upgrade or patch preparation, the Upgrader service validates the currently installed appliance software version against the new target version. The preparation process (`upgradePreConfig`) documented in this section will only proceed if validation is successful.
If the appliance is not running at least the minimum required version, the Upgrader exits the process and rolls back the environment to its previous state. View the details of the failed upgrade job:

```
PCA-ADMIN> getupgradejob upgradeJobId=1700153626051-prepare-40046
Data:
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_prepare_environment_2023_11_16-16.53.46.log
  Arguments = [...]
  Status = Failed
[...]
  Tasks 23 - Name = Check Prerequisite Build Version
  Tasks 23 - Description = Check current build version not lower than
prerequisite version
  Tasks 24 - Name = Check Prerequisite Build Version
  Tasks 24 - Message = (("Caught exception while checking prerequisite build
number
  Exception: Command: ['/usr/bin/python3', '/var/lib/pca-upgrader/
prerequisite_build_validator.py',
  'rack=PCA', 'upgrade=ISO'] failed (1): stderr: b'' stdout: b'PCA version is
lower than prerequisite build,
  must upgrade to prerequisite build 3.0.2-b799577 to proceed further upgrade\
\n'",), {})
  Tasks 24 - Status = Failed
```

You must first install the prerequisite version, which is indicated by the error message in the upgrade job output.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: Upgrade PreConfig.
4. Fill out the upgrade request parameters:
   - **Option:** Enter *ISO*.
   - **Location:** Enter the path to the location where the ISO image is stored.
   - **ISO Checksum:** Enter the checksum required to verify the integrity of the ISO image. The checksum is provided alongside the ISO image in a file named `<iso_image>.sha256sum`.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Alternative ULN Channel:** This parameter applies to patching and can be ignored.
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

   The ISO image is unpacked on the shared storage of the management node cluster, from where its contents can be used to perform the required upgrade operations.

Using the Service CLI

1. Gather the information that you need to run the command:
   - the location of the ISO image to upgrade from

     Enter the path to where the ISO image is stored. Its contents will be unpacked on the shared storage accessible from the management nodes.
   - the checksum used to verify the ISO image

     The checksum is provided alongside the ISO image. Its file name is the ISO image name with `.sha256sum` appended. The system uses the checksum to verify that the data in the ISO image is intact and valid for this upgrade.
2. Enter the upgrade pre-configuration command.

   Syntax (entered on a single line):

   ```
   upgradePreConfig
   option=ISO
   location=<path-to-iso>
   isoChecksum=<iso-file-checksum>
   ```

   Example:

   ```
   PCA-ADMIN> upgradePreConfig  option=ISO \
   location="http://host.example.com/pca-<version>-<build>.iso" \
   isoChecksum=90e4505b098031afb02068080db2603dc6f580cd7cf52aa51ecd0c3b81668027
   Command: upgradePreConfig  option=ISO location="http://host.example.com/pca-
   <version>-<build>.iso"
   ```

```
isoChecksum=90e4505b098031afb02068080db2603dc6f580cd7cf52aa51ecd0c3b81668027
Status: Success
Time: 2022-11-06 06:35:38,884 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1668417666968-
prepare-28142 Upgrade Request Id = UWS-c94ba56a-1b91-49d8-8e51-afeae7f62186
```

3. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id
upgradeRequestId                          commandName   result
  --
----------------                          -----------   ------
  1630938939109-compute-7545
UWS-61736806-7e5a-4648-9259-07c54c39cacb   compute       Passed
  1632849609034-kubernetes-35545   UWS-edfa3b32-
c32a-4b67-8df5-2357096052bf   kubernetes    Passed
  1668417666968-prepare-28142      UWS-c94ba56a-1b91-49d8-8e51-
afeae7f62186   prepare       Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1668417666968-prepare-28142
Command: getUpgradeJob upgradeJobId=1668417666968-prepare-28142
Status: Success
Time: 2022-11-06 07:24:00,793 UTC
Data:
  Upgrade Request Id = UWS-c94ba56a-1b91-49d8-8e51-afeae7f62186
  Name = prepare
  Start Time = 2022-06-14T06:35:56
  End Time = 2022-11-06T06:35:58
  Pid = 28142
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_prepare_environment_2022_11_16-06.35.56.log
  Arguments =
{"component_names":null,"diagnostics":false,"display_task_plan":false,"dry_ru
n_tasks":false,"expected_iso_checksum":null,"fail_halt":false,"fail_upgrade":
null,"image_location":"http://host.example.com/pca-<version>-
<build>.iso","online_upgrade":null,"precheck_status":false,"repo_config_overr
ide":null,"result_override":null,"task_time":0,"test_run":false,"upgrade":fal
se,"upgrade_to":null,"user_uln_base_url":null,"verify_only":false,"host_ip":n
ull,"log_level":null,"switch_type":null,"epld_image_location":null,"checksum"
:"90e4505b098031afb02068080db2603dc6f580cd7cf52aa51ecd0c3b81668027","composit
ion_id":null,"request_id":"UWS-c94ba56a-1b91-49d8-8e51-
afeae7f62186","uln":null,"patch":null}
  Status = Passed
  Execution Time(sec) = 616
  Tasks 1 - Name = Validate Image Location
  Tasks 1 - Description = Verify that the image exists at the specified
location and is correctly named
[...]
```

4. Proceed to the next upgrade preparation phase.

Backup Before Upgrade

For system-critical components and services, Oracle Private Cloud Appliance runs a scheduled backup service that allows the appliance to be restored to its last known healthy state in case of a catastrophic failure. It is recommended that you implement a backup strategy for the users' cloud resources in the Compute Enclave as well.

Before upgrading any component of the Private Cloud Appliance, you should create a backup of the latest state of the MySQL database, the ZFS Storage Appliance and the Secret Service (Vault). The backup commands leverage the existing backup service but create an additional restore point that includes the most recent changes from right before you start an upgrade.

Using the Service CLI

1. Start the three required backup tasks.

```
PCA-ADMIN> backup target=vault
Command: backup target=vault
Status: Success
Time: 2022-11-06 09:56:18,786 UTC
Data:
  Type = BackupJob
  Job Id = ocid1.brs-
job.PCA3X62D9C1.mypca.joopwuv9403uzbfrh4x9mprmoduh3ljais6ex233v1b21ccqywu4a3vqykgm
  Display Name = brs-job-1668419778-backup
  Profile Id =
ocid1.backup_profile.PCA3X62D9C1.mypca.wrxfwtxwxw6ydp2mwnypcaaxxzmwpuhsc33gcm3dyte7
kgr4etuhb29qbs8q
  Time Created = 2022-11-06T09:56:18Z
  Lifecycle State = CREATING
  Retention = 14

PCA-ADMIN> backup target=zfs
Command: backup target=zfs
Status: Success
Time: 2022-11-06 09:57:23,084 UTC
Data:
  Type = BackupJob
  Job Id = ocid1.brs-
job.PCA3X62D9C1.mypca.9oaeaa2kw5crqfcjkh8kyhbxcv8bwh0f4ud6n3lucf802oj15ss3k39874bc
  Display Name = brs-job-1668419842-backup
  Profile Id =
ocid1.backup_profile.PCA3X62D9C1.mypca.p7w0tgbvhtjqsgc8rllca2cvotkpgrtf4huiph7466mj
io0dgskij9f0bp06
  Time Created = 2022-11-06T09:57:22Z
  Lifecycle State = CREATING
  Retention = 14

PCA-ADMIN> backup target=mysql
Command: backup target=mysql
Status: Success
Time: 2022-11-06 09:57:30,229 UTC
Data:
  Type = BackupJob
  Job Id = ocid1.brs-
job.PCA3X62D9C1.mypca.iew5tphpgr3h6mhliw2fai2ywvv386a0xc7isfo8kisj0wrcx114irnit6ot
  Display Name = brs-job-1668419850-backup
  Profile Id =
ocid1.backup_profile.PCA3X62D9C1.mypca.henfqzdbafs4z3mxeuslb1c6f4t049w0pxvwf1gi3eb8
wm1y11v7m932tn4g
  Time Created = 2022-11-06T09:57:30Z
  Lifecycle State = CREATING
  Retention = 14
```

2. Use the backup job ID to check the status of the backups.

```
PCA-ADMIN> getBackupJobs
Command: getBackupJobs
Status: Success
```

```
Time: 2022-11-06 10:03:18,986 UTC
Data:

id
                    displayName                 components

--
                    -----------                 ----------
  ocid1.brs-
job.PCA3X62D9C1.mypca.iew5tphpgr3h6mhliw2fai2ywvv386a0xc7isfo8kisj0wrcx114irn
it6ot   brs-job-1668419850-backup   mysql
  ocid1.brs-
job.PCA3X62D9C1.mypca.9oaeaa2kw5crqfcjkh8kyhbxcv8bwh0f4ud6n3lucf802oj15ss3k39
874bc   brs-job-1668419842-backup   zfs
  ocid1.brs-
job.PCA3X62D9C1.mypca.joopwuv9403uzbfrh4x9mprmoduh3ljais6ex233v1b21ccqywu4a3v
qykgm   brs-job-1668419778-backup   vault

PCA-ADMIN> getBackupJob backupJobId=ocid1.brs-
job.PCA3X62D9C1.mypca.iew5tphpgr3h6mhliw2fai2ywvv386a0xc7isfo8kisj0wrcx114irn
it6ot
Command: getBackupJob backupJobId=ocid1.brs-
job.PCA3X62D9C1.mypca.iew5tphpgr3h6mhliw2fai2ywvv386a0xc7isfo8kisj0wrcx114irn
it6ot
Status: Success
Time: 2022-11-06 10:04:07,080 UTC
Data:
  Type = BackupJob
  Job Id = ocid1.brs-
job.PCA3X62D9C1.mypca.iew5tphpgr3h6mhliw2fai2ywvv386a0xc7isfo8kisj0wrcx114irn
it6ot
  Display Name = brs-job-1668419850-backup
  Time Created = 2022-11-06T09:57:30Z
  Status = success
  Components = mysql
```

3. Confirm that all three backup operations have completed successfully. Then, proceed to the next upgrade preparation phase.

Upgrade the Upgrader

The code of the upgrader is changed regularly, as is the case with any other system service. Ensure that the upgrader is up-to-date first, so that all upgrade commands are run with the latest version.

Upgrade operations will not be allowed to run if the system detects that the Upgrader is not the latest available version.

In software version 3.0.2-b892153 and later, when the Upgrader is upgraded to the latest version, an *upgrade plan* is also generated. The upgrade plan is based on a comparison between the current installation and the packages downloaded during the preparation of the upgrade environment, and it determines which upgrades need to be performed in the next phase. For more information, see Checking Upgrade Plan Status and Progress.

> **Caution:**
>
> Ensure that the Upgrade Preconfig task has been completed first. See Preparing the Upgrade Environment.
>
> To prevent inconsistencies while executing the upgrade plan later on, it is critical that both parts of the preparation process, Upgrade PreConfig and preUpgrade, are completed together in the specified order. If at any time you need to rerun the preUpgrade command, you must rerun the preceding command first.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: preUpgrade.
4. Fill out the upgrade request parameters:
   - **Action:** Enter *start* to retrieve the new version of the upgrader. (To check the status of the previous preUpgrade job, enter *status* instead.)
   - **Type:** Enter *ISO*. The upgrader packages are picked from the unpacked ISO image.
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table. When the job has completed successfully, the upgrader is up-to-date and ready for use.

Using the Service CLI

1. Start the process to install the latest upgrader version on the management nodes.

```
PCA-ADMIN> preUpgrade action=start type=ISO
Data:
  Successfully triggered the pre-upgrade task.
   Pre upgrade status = success
```

   As part of the process, these operations are performed:

   a. Save the existing yum configuration.
   b. Configure the yum repository for the new upgrader files.
   c. Install the new upgrader version on the management nodes, then restart the upgrader systemd service for the changes to take effect.
   d. Restore the existing yum configuration that was saved in the first step.
2. Check the status of the upgrade process at any time using this command:

```
PCA-ADMIN> preUpgrade action=status
Data:
  A pre-upgrade task is running!
  Pre upgrade status = IN-PROGRESS

PCA-ADMIN> preUpgrade action=status
Data:
```

```
The previous pre-upgrade task succeeded!
Pre upgrade status = SUCCESS
```

3. Confirm that the latest version of the upgrader has been installed successfully. Ensure that the system is in ready state. Then, proceed with the component upgrades.
4. Optionally, check which components need to be upgraded by displaying the upgrade plan.

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

```
PCA-ADMIN> getUpgradePlan
Data:
  id                        componentType   currentBuild
targetBuild      currentVersion
targetVersion                  requireReboot   timeEstimation
requireUpgrade   impactedInfra
  --                        -------------   ------------
-----------      --------------
-------------                  -------------   --------------
--------------   -------------
  generic                   zfssa           3.0.2-b892153    3.0.2-
b1052646   2013.06.05.8.57.1-2.57.5392.2   2013.06.05.8.57.1-2.57.5501.4
false           45              true            host,compute
  100.96.2.64               compute         3.0.2-b892153    3.0.2-
b1052646   3.0.2-640                       3.0.2-691
true            20              true            compute
  100.96.2.65               compute         3.0.2-b892153    3.0.2-
b1052646   3.0.2-640                       3.0.2-691
true            20              true            compute
  100.96.2.66               compute         3.0.2-b892153    3.0.2-
b1052646   3.0.2-640                       3.0.2-691
true            20              true            compute
  100.96.2.67               compute         3.0.2-b892153    3.0.2-
b1052646   3.0.2-640                       3.0.2-691
true            20              true            compute
  100.96.2.68               compute         3.0.2-b892153    3.0.2-
b1052646   3.0.2-640                       3.0.2-691
true            20              true            compute
  100.96.2.33               host            3.0.2-b892153    3.0.2-
b1052646   oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
true            35              true            host
  100.96.2.34               host            3.0.2-b892153    3.0.2-
b1052646   oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
true            35              true            host
  100.96.2.35               host            3.0.2-b892153    3.0.2-
b1052646   oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
true            35              true            host
  generic                   mysql           3.0.2-b892153    3.0.2-
b1052646   8.0.28-1.1                      8.0.33-1.1
false           15              true            host
  generic                   etcd            3.0.2-b892153    3.0.2-
b1052646   3.3.10                          3.3.10
false           5               false           host
```

```
  generic                    vault           3.0.2-b892153    3.0.2-b1052646
v1.7.1-3                       v1.7.1-3                         false
5                 false          host
  generic                    kubernetes      3.0.2-b892153    3.0.2-b1052646
1.20.6-1                       1.25.7-1                         false
350               true           host,compute
  generic                    platform        3.0.2-b892153    3.0.2-b1052646
None                           None                             false
50                true           host,compute
  Oracle-Linux-7.9           ociImages       3.0.2-b892153    3.0.2-b1052646
2022.08.29_0                   2023.09.26_0                     false
5                 true           host
  Oracle-Linux-8             ociImages       3.0.2-b892153    3.0.2-b1052646
2022.08.29_0                   2023.09.26_0                     false
5                 true           host
  Oracle-Linux-9             ociImages       3.0.2-b892153    3.0.2-b1052646
None                           2023.09.26_0                     false
5                 true           host
  Oracle-Linux8-OKE-1.26.6   ociImages       3.0.2-b892153    3.0.2-b1052646
None                           20240210                         false
5                 true           host
  Oracle-Linux8-OKE-1.27.7   ociImages       3.0.2-b892153    3.0.2-b1052646
None                           20240209                         false
5                 true           host
  Oracle-Linux8-OKE-1.28.3   ociImages       3.0.2-b892153    3.0.2-b1052646
None                           20240210                         false
5                 true           host
  Oracle-Solaris-11          ociImages       3.0.2-b892153    3.0.2-b1052646
2023.04.18_0                   2023.10.16_0                     false
5                 true           host
  100.96.0.33                ilom            3.0.2-b892153    3.0.2-b1052646
5.0.2.23                       5.1.1.21                         true
10                true           host,compute
  100.96.0.34                ilom            3.0.2-b892153    3.0.2-b1052646
5.0.2.23                       5.1.1.21                         true
10                true           host,compute
  100.96.0.35                ilom            3.0.2-b892153    3.0.2-b1052646
5.0.2.23                       5.1.1.21                         true
10                true           host,compute
  100.96.0.64                ilom            3.0.2-b892153    3.0.2-b1052646
5.0.2.23                       5.1.2.20.a                       true
10                true           host,compute
  100.96.0.65                ilom            3.0.2-b892153    3.0.2-b1052646
5.0.2.23                       5.1.1.21                         true
10                true           host,compute
  100.96.0.66                ilom            3.0.2-b892153    3.0.2-b1052646
5.0.2.23                       5.1.2.21                         true
10                true           host,compute
  100.96.0.67                ilom            3.0.2-b892153    3.0.2-b1052646
5.0.2.23                       5.1.2.20.a                       true
10                true           host,compute
  100.96.0.68                ilom            3.0.2-b892153    3.0.2-b1052646
5.0.2.23                       5.1.2.21                         true
10                true           host,compute
  leaf                       switch          3.0.2-b892153    3.0.2-b1052646
10.2.5                         10.3.4a                          false
60                true           host,compute
  mgmt                       switch          3.0.2-b892153    3.0.2-b1052646
10.2.5                         10.3.4a                          false
60                true           host,compute
  spine                      switch          3.0.2-b892153    3.0.2-b1052646
```

```
    10.2.5                          10.3.4a
    false           60               true             host,compute
```

Checking Current Version of Components

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

To evaluate upgrade requirements and the impact of an upgrade you can view the current state of the top-level rack components using a convenient Service CLI command. It lists the most important build and version numbers of all the components for which an upgrade procedure is documented in this guide.

Log in to the Service CLI and execute this command:

```
PCA-ADMIN> getComponentVersions
Data:
  id                          component    iso              version
  --                          ---------    ---              -------
  100.96.2.64                 compute      3.0.2-b1046481   3.0.2-687
  100.96.2.65                 compute      3.0.2-b1046481   3.0.2-687
  100.96.2.66                 compute      3.0.2-b1046481   3.0.2-687
  100.96.2.67                 compute      3.0.2-b1046481   3.0.2-687
  100.96.2.68                 compute      3.0.2-b1046481   3.0.2-687
  generic                     etcd         3.0.2-b1049367   3.3.10
  100.96.2.33                 host         3.0.2-b1049367   oraclelinux-
release-7.9-1.0.9
  100.96.2.34                 host         3.0.2-b1049367   oraclelinux-
release-7.9-1.0.9
  100.96.2.35                 host         3.0.2-b1049367   oraclelinux-
release-7.9-1.0.9
  100.96.0.33                 ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.34                 ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.35                 ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.64                 ilom         3.0.2-b1049367   5.1.2.20.a
  100.96.0.65                 ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.66                 ilom         3.0.2-b1049367   5.1.1.21
  100.96.0.67                 ilom         3.0.2-b1049367   5.1.2.20.a
  100.96.0.68                 ilom         3.0.2-b1049367   5.1.1.21
  generic                     kubernetes   3.0.2-b1049367   1.25.15-1
  generic                     mysql        3.0.2-b1049367   8.0.33-1.1
  Oracle-Linux-7.9            ociImages    3.0.2-b1049367   2023.09.26_0
  Oracle-Linux-8              ociImages    3.0.2-b1049367   2023.09.26_0
  Oracle-Linux-9              ociImages    3.0.2-b1049367   2023.09.26_0
  Oracle-Linux8-OKE-1.26.6    ociImages    3.0.2-b1049367   20240210
  Oracle-Linux8-OKE-1.27.7    ociImages    3.0.2-b1049367   20240209
  Oracle-Linux8-OKE-1.28.3    ociImages    3.0.2-b1049367   20240210
  Oracle-Solaris-11           ociImages    3.0.2-b1049367   2023.10.16_0
  generic                     platform     3.0.2-b1046481   None
  leaf                        switch       3.0.2-b1049367   10.3.4a
  mgmt                        switch       3.0.2-b1049367   10.3.4a
  spine                       switch       3.0.2-b1049367   10.3.4a
  generic                     vault        3.0.2-b1049367   v1.7.1-3
```

```
  generic                    zfssa        3.0.2-b1046481
2013.06.05.8.57.1-2.57.5501.1
```

> **Note:**
>
> Alternatively, you can view this information in the Service Web UI. In the navigation menu, go to Maintenance and select Component Version.

Ensuring the System Is In Ready State

Upgrades can be performed with limited impact on the system. No downtime is required, and user workloads continue to run while the underlying infrastructure is being upgraded in stages. However, it is considered good practice to ensure that backups are created of the system and the resources in your environment.

Every upgrade operation is preceded by a set of pre-checks. These are built into the upgrade code and will report an error if the system is not in the required state for the upgrade. The upgrade will only begin if all pre-checks are passed.

You can use the pre-checks to test in advance for any system health issues that would prevent a successful upgrade. After preparing the upgrade environment, run any or all of the upgrade commands with the "verify only" option.

> **Caution:**
>
> Oracle strongly recommends testing that the Private Cloud Appliance is ready for upgrading, by executing the full management node upgrade command in verify-only mode. The output provides a readiness report you can use to plan any corrective actions as well as the upgrade.

In the Service Web UI the verify-only option is activated with a check box when you create the upgrade request; in the Service CLI you use the optional upgrade command parameter shown in this example:

```
PCA-ADMIN> upgradeKubernetes verifyOnly=True
[...]

PCA-ADMIN> getUpgradeJobs
  id                                        upgradeRequestId
commandName          result
  --                                        ----------------
-----------          ------
  1632849609034-kubernetes_verify-35545   UWS-8995e5b7-a237-4717-bb5c-01f1cf85daf0
kubernetes_verify    Passed
```

If issues are detected, you can resolve them before the planned upgrade window, and keep the actual system upgrade as fluent and short as possible.

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

It is important to note that concurrent upgrade operations are not supported. An upgrade job must be completed before a new one can be started.

3
Checking Upgrade Plan Status and Progress

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

The Upgrader uses an *upgrade plan* as a kind of checklist to perform all upgrade operations, which implies full management cluster upgrades as well as individual component upgrades. The Oracle Private Cloud Appliance Concepts Guide describes this approach in more detail as part of the "Upgrade" section in the chapter Appliance Administration Overview.

The upgrade plan is generated when the Upgrader itself is upgraded to the latest version. The plan is based on a comparison of the currently installed components on the rack, and the target component versions and latest packages downloaded to shared storage during the preparation of the upgrade environment. The resulting upgrade plan shows for which components an upgrade procedure needs to be executed in the next phase.

All components must be upgraded in a prescribed order. The upgrade plan will prevent a component upgrade procedure from starting if the preceding upgrades have not been completed. An error message informs the administrator which components need to be upgraded first.

> **Note:**
>
> In general, firmware may be upgraded whenever new versions are made available for your system. Firmware upgrades can be applied in no particular order and independently of other components.
>
> However, there is an extra requirement: when upgrading to appliance software version 3.0.2-b1081557 or later, the ZFS Storage Appliance firmware must be upgraded before all other components.

This is the order of operations enforced through the upgrade plan:

1. Prepare upgrade environment (Upgrade PreConfig)
2. Upgrade the Upgrader (PreUpgrade)
3. ZFS Storage Appliance firmware (version 3.0.2-b1081557 or later)
4. Compute nodes
5. Host operating system of management nodes
6. MySQL cluster database
7. Secret service (including Etcd and Vault)

8. Kubernetes container orchestration packages (platform layer)
9. Containerized microservices
10. Oracle Cloud Infrastructure images

Once the upgrade environment has been prepared, all upgrade operations required to bring the system up-to-date are listed in the upgrade plan. Whenever an upgrade procedure has been completed successfully, the upgrade plan is updated with the latest status: for upgraded components the source and target versions are identical and the "upgrade required" flag is disabled.

At any point in time you can check how far the system has progressed through the upgrade plan. It indicates which components are already up-to-date and which still require upgrading.

The upgrade plan can be viewed in two ways:

- In the Service Web UI, go to Maintenance in the navigation menu and select Upgrade Plan.
- In the Service CLI, enter the following command:

```
PCA-ADMIN> getUpgradePlan
Data:
  id                        component    currentBuild     targetBuild
currentVersion                 targetVersion
requireReboot   timeEstimation (min)   requireUpgrade   impactedInfra
  --                        ---------    ------------     -----------
--------------                 -------------
-------------   --------------------   --------------   -------------
  generic                   zfssa        3.0.2-b1053709   3.0.2-b1053709
2013.06.05.8.57.1-2.57.5501.4   2013.06.05.8.57.1-2.57.5501.4
false           45                     false            host,compute
  100.96.2.64               compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                      3.0.2-691
false           20                     false            compute
  100.96.2.65               compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                      3.0.2-691
false           20                     false            compute
  100.96.2.66               compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                      3.0.2-691
false           20                     false            compute
  100.96.2.67               compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                      3.0.2-691
false           20                     false            compute
  100.96.2.68               compute      3.0.2-b1053709   3.0.2-b1053709
3.0.2-691                      3.0.2-691
false           20                     false            compute
  100.96.2.33               host         3.0.2-b1053709   3.0.2-b1053709
oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
false           35                     false            host
  100.96.2.34               host         3.0.2-b1053709   3.0.2-b1053709
oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
false           35                     false            host
  100.96.2.35               host         3.0.2-b1053709   3.0.2-b1053709
oraclelinux-release-7.9-1.0.9   oraclelinux-release-7.9-1.0.9
false           35                     false            host
  generic                   mysql        3.0.2-b1053709   3.0.2-b1053709
8.0.33-1.1                     8.0.33-1.1
false           15                     false            host
  generic                   etcd         3.0.2-b1053709   3.0.2-b1053709
```

```
3.3.10                          3.3.10                          false
5                       false           host
  generic                       vault       3.0.2-b1053709  3.0.2-b1053709
v1.7.1-3                        v1.7.1-3                        false
5                       false           host
  generic                       kubernetes  3.0.2-b1053709  3.0.2-b1053709
1.25.7-1                        1.25.7-1                        false
80                      false           host,compute
  generic                       platform    3.0.2-b1053709  3.0.2-b1053709
None                            None                            false
50                      false           host,compute
  Oracle-Linux-7.9              ociImages   3.0.2-b1053709  3.0.2-b1053709
2023.09.26_0                    2023.09.26_0                    false
5                       false           host
  Oracle-Linux-8                ociImages   3.0.2-b1053709  3.0.2-b1053709
2023.09.26_0                    2023.09.26_0                    false
5                       false           host
  Oracle-Linux-9                ociImages   3.0.2-b1053709  3.0.2-b1053709
2023.09.26_0                    2023.09.26_0                    false
5                       false           host
  Oracle-Linux8-OKE-1.26.6      ociImages   3.0.2-b1053709  3.0.2-b1053709
20240210                        20240210                        false
5                       false           host
  Oracle-Linux8-OKE-1.27.7      ociImages   3.0.2-b1053709  3.0.2-b1053709
20240209                        20240209                        false
5                       false           host
  Oracle-Linux8-OKE-1.28.3      ociImages   3.0.2-b1053709  3.0.2-b1053709
20240210                        20240210                        false
5                       false           host
  Oracle-Solaris-11             ociImages   3.0.2-b1053709  3.0.2-b1053709
2023.10.16_0                    2023.10.16_0                    false
5                       false           host
  100.96.0.33                   ilom        3.0.2-b1053709  3.0.2-b1053709
5.1.1.21                        5.1.1.21                        false
10                      false           host,compute
  100.96.0.34                   ilom        3.0.2-b1053709  3.0.2-b1053709
5.1.1.21                        5.1.1.21                        false
10                      false           host,compute
  100.96.0.35                   ilom        3.0.2-b1053709  3.0.2-b1053709
5.1.1.21                        5.1.1.21                        false
10                      false           host,compute
  100.96.0.64                   ilom        3.0.2-b1053709  3.0.2-b1053709
5.1.2.20.a                      5.1.2.20.a                      false
10                      false           host,compute
  100.96.0.65                   ilom        3.0.2-b1053709  3.0.2-b1053709
5.1.1.21                        5.1.1.21                        false
10                      false           host,compute
  100.96.0.66                   ilom        3.0.2-b1053709  3.0.2-b1053709
5.1.1.21                        5.1.1.21                        false
10                      false           host,compute
  100.96.0.67                   ilom        3.0.2-b1053709  3.0.2-b1053709
5.1.2.20.a                      5.1.2.20.a                      false
10                      false           host,compute
  100.96.0.68                   ilom        3.0.2-b1053709  3.0.2-b1053709
5.1.1.21                        5.1.1.21                        false
10                      false           host,compute
  leaf                          switch      3.0.2-b926028   3.0.2-b1053709
10.2.3                          10.3.4a                         false
60                      true            host,compute
  mgmt                          switch      3.0.2-b926028   3.0.2-b1053709
10.2.3                          10.3.4a                         false
```

```
60                      true              host,compute
  spine                       switch       3.0.2-b926028    3.0.2-b1053709
10.2.3                          10.3.4a
false           60                      true             host,compute
```

4
Checking System Upgrade History

Information about all component upgrades and patches is stored in upgrade jobs, which can be consulted in the Service Web UI and Service CLI. Over time, as the system goes through multiple upgrades, the large number of entries might make the list difficult to interpret. The upgrade history provides a clear way to drill down into the details of the upgrade and patching activity on your appliance.

The upgrade history presents the information from all upgrade and patch jobs in a categorized way so you can see which version upgrades have been performed, which jobs have been run for each of those upgrades, and from which source (ISO upgrade or ULN patch). Details include build versions, component versions before and after, job completion, success or failure, time stamps, and duration.

Appliance software builds are installed onto the appliance either through upgrade from an ISO image or patching from ULN. Display the build history by running the following command.

```
PCA-ADMIN> GetUpgradeHistory
Data:
  id                                    From Build       To Build          Type
Status         Start Time               End Time               Actual Upgrade
Time(min)   Total Upgrade Time(min)
  --                                    -----            -------           -----
--------       ----------               --------
--------------             ------------
  pca-upgrade-history-3.0.2-b951413    3.0.2-b868711    3.0.2-b951413     ISO
Incomplete     2023-06-14T10:32:14      2023-06-14T15:32:14
300                        600
  pca-upgrade-history-3.0.2-b868711    3.0.2-b854356    3.0.2-b868711     ULN
Completed      2023-01-01T15:10:07      2023-01-01T19:10:07
250                        500
  pca-upgrade-history-3.0.2-b854356    3.0.2-b790137    3.0.2-b854356     ISO
Completed      2022-12-14T06:01:00      2022-12-14T12:01:00
350                        700
```

If necessary, you can filter results by type.

```
PCA-ADMIN> GetUpgradeHistory type=uln
Data:
  id                                    From Build       To Build          Type
Status         Start Time               End Time               Actual Upgrade
Time(min)   Total Upgrade Time(min)
  --                                    -----            -------           -----
--------       ----------               --------
--------------             ------------
  pca-upgrade-history-3.0.2-b868711    3.0.2-b854356    3.0.2-b868711     ULN
Completed      2023-01-01T15:10:07      2023-01-01T19:10:07
250                        500
```

To display the upgrade job list related to a particular build, copy the build ID and run this command:

```
PCA-ADMIN> GetUpgradeHistoryDetails id=pca-upgrade-history-3.0.2-b951413
Data:
```

```
  id           component  Timestamp             From Version
To Version                      Status   Job ID
Time Taken(min)
  --           ---------  ---------             ------------
----------                      -------  -------
----------
  spine        cisco      2023-06-14T10:32:14   9.3.2
10.2.3                          Failed   1686766376945-cisco-1509
50
  mgmt         cisco      2023-06-14T10:22:14   9.3.2
10.2.3                          Passed   1686762287214-cisco-31252
60
  leaf         cisco      2023-06-14T10:12:14   9.3.2
10.2.3                          Passed   1686758831077-cisco-35671
45
  generic      zfssa      2023-06-14T09:52:14   2013.06.05.8.40.1-2.40.4958.31
2013.06.05.8.48.1-2.48.5222.1   Passed   1686768921264-zfssa-19292
30
  100.96.0.66  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19293
55
  100.96.0.65  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19292
50
  100.96.0.64  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19294
60
  100.96.0.35  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19295
35
  100.96.0.34  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19296
30
  100.96.0.33  ilom       2023-06-14T09:22:14   5.0.2.23
5.1.1.21                        Passed   1686768921264-ilom-19297
65
  generic      platform   2023-06-14T09:12:14   None
None                            Passed   1686552138506-platform-48794
100
  generic      kubernetes 2023-06-14T09:12:14   1.20.6-1
1.25.7-1                        Passed   1685379294807-kubernetes-72808
40
  generic      vault      2023-06-14T09:12:14   v1.7.1-3
v1.7.1-3                        Passed   1685614045193-vault-47652
63
  generic      etcd       2023-06-14T09:02:14   3.3.10
3.3.10                          Passed   1685613743232-etcd-83924
50
  generic      mysql      2023-06-14T08:52:14   8.0.28-1.1
8.0.30-1.1                      Passed   1685378389035-mysql-90009
45
  100.96.2.35  host       2023-06-14T08:42:14   oraclelinux-release-7.9-1.0.9
oraclelinux-release-7.9-1.0.9   Passed   1686726229799-host-68919
20
  100.96.2.34  host       2023-06-14T08:32:14   oraclelinux-release-7.9-1.0.9
oraclelinux-release-7.9-1.0.9   Passed   1686726229799-host-68919
30
  100.96.2.33  host       2023-06-14T08:22:14   oraclelinux-release-7.9-1.0.9
oraclelinux-release-7.9-1.0.9   Passed   1686726229799-host-68919
25
  100.96.2.66  compute    2023-06-14T07:52:14   3.0.2-502
```

```
3.0.2-630                       Passed   1685607717395-compute-53331          50
  100.96.2.65  compute    2023-06-14T07:42:14   3.0.2-502
3.0.2-630                       Passed   1685606912300-compute-91673          66
  100.96.2.64  compute    2023-06-14T07:32:14   3.0.2-502
3.0.2-630                       Passed   1685372050358-compute-50568          30
  100.96.2.64  compute    2023-06-14T07:22:14   3.0.2-502
3.0.2-630                       Failed   1685372050358-compute-50560          45
  generic      preupgrade 2023-06-14T06:32:14   3.0.2-b868711                3.0.2-
b951413                    N/A      N/A                                 N/A
```

5
Upgrading a Compute Node

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.
>
> When upgrading to appliance software version 3.0.2-b1081557 or later, the ZFS Storage Appliance firmware must be upgraded before all other components. For more information, see Checking Upgrade Plan Status and Progress.

The compute node upgrade ensures that the latest Oracle Linux kernel and user space packages are installed, as well as the `ovm-agent` package with appliance-specific optimizations. Compute nodes must be locked and upgraded one at a time; concurrent upgrades are not supported. After successful upgrade, when a compute node has rebooted, the administrator must manually remove the locks to allow the node to return to normal operation.

> **Note:**
>
> In case the ILOM also needs to be upgraded, you can integrate it into this procedure by executing the optional steps. The combined procedure eliminates the need to evacuate and reboot the same node twice.

> **Note:**
>
> In software versions 3.0.2-b892153 and later the Upgrader service uses the upgrade plan, generated during the pre-upgrade process, to determine whether a component needs to be upgraded. If a component is already at the required version, the upgrade command does start an upgrade job, but it is completed immediately because the upgrade plan indicates there is nothing to do.
>
> Practically speaking, when a component is already at the required version, the upgrade procedure is skipped. However, a same-version upgrade can be forced using the Service Web UI or Service CLI command option, if necessary. For example: `upgradeCN hostIp=100.96.2.64 force=True`.

To obtain the host IP address of a compute node, use one of these Service CLI commands:

- Enter `show ComputeNode name=<node_name>` and look for the `Ip Address` in the output.

```
PCA-ADMIN> show ComputeNode name=pcacn001
[...]
```

```
Data:
  Id = cfbc27ce-f682-4a50-9299-0dc939fbb6be
  Type = ComputeNode
  Provisioning State = Provisioned
  [...]
  HW Id = 2146XLD02G
  MAC Address = 00:10:e0:fe:00:58
  Ip Address = 100.96.2.64
  ILOM Ip Address = 100.96.0.64
  ILOM MAC Address = 00:10:e0:fe:00:5b
  [...]
```

- On systems running software version 3.0.2-b892153 or later, enter `getServerIP hostName=<node_name>`.

```
PCA-ADMIN> getServerIP hostName=pcacn001
[...]
Data:
  status = success
  data = 100.96.2.64
```

Using the Service Web UI

1. Set the provisioning and maintenance locks for the compute node you are about to upgrade. Ensure that no active compute instances are present on the node.

> **Caution:**
>
> Depending on the high-availability configuration of the Compute service, automatic instance migrations can prevent you from successfully locking a compute node. For more information, refer to the following sections in the Hardware Administration chapter of the Oracle Private Cloud Appliance Administrator Guide:
>
> - Migrating instances and locking a compute node: see "Performing Compute Node Operations".
> - Compute service HA configuration: see "Configuring the Compute Service for High Availability".

   a. In the navigation menu, click Rack Units. In the Rack Units table, click the name of the compute node you want to upgrade to display its detail page.

   b. In the top-right corner of the compute node detail page, click Controls and select the Provisioning Lock command.

   c. When the provisioning lock has been set, click Controls again and select the Maintenance Lock command. This command might fail if instance migrations are in progress. Wait a few minutes and retry.

2. In the navigation menu, click Upgrade & Patching.
3. Optionally, upgrade the server ILOM first.

   a. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch. The Create Request window appears.

   b. Choose *Upgrade* as the Request Type. Select the appropriate upgrade request type: Upgrade ILOM.

Fill out the server's assigned IP address in the ILOM network. This is an IP address in the internal 100.96.0.0/23 range.

c. Click Create Request. The new upgrade request appears in the Upgrade Jobs table.

d. Wait 5 minutes to allow the ILOM upgrade job to complete. Then proceed to the host upgrade.

4. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.

5. Select the appropriate upgrade request type: Upgrade CN.

6. Fill out the upgrade request parameters:

   - **Host IP:** Enter the compute node's assigned IP address in the internal administration network. This is an IP address in the internal 100.96.2.0/23 range.
   - **Image Location:** Enter the path to the location where the ISO image is stored. This parameter is deprecated in software version 3.0.2-b892153 and later.
   - **ISO Checksum:** Enter the checksum to verify the ISO image. It is stored alongside the ISO file. This parameter is deprecated in software version 3.0.2-b892153 and later.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.

7. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

8. When the compute node has been upgraded successfully, release the provisioning and maintenance locks.

   For more information, refer to the section "Performing Compute Node Operations". It can be found in the chapter Hardware Administration of the Oracle Private Cloud Appliance Administrator Guide.

   a. Open the compute node detail page.

   b. In the top-right corner of the compute node detail page, click Controls and select the Maintenance Unlock command.

   c. When the maintenance lock has been released, click Controls again and select the Provisioning Unlock command.

Using the Service CLI

1. Get the IP address of the compute node you intend to upgrade.
2. Set the provisioning and maintenance locks for the compute node you are about to upgrade.

> **Caution:**
>
> Depending on the high-availability configuration of the Compute service, automatic instance migrations can prevent you from successfully locking a compute node. For more information, refer to the following sections in the Hardware Administration chapter of the Oracle Private Cloud Appliance Administrator Guide:
>
> - Migrating instances and locking a compute node: see "Performing Compute Node Operations".
> - Compute service HA configuration: see "Configuring the Compute Service for High Availability".

```
PCA-ADMIN> list ComputeNode
Data:
  id                                     name       provisioningState
provisioningType
  --                                     ----       -----------------
----------------
  363a26f4-fa34-4e4c-8e17-a1671a0b77d1   pcacn001   Provisioned         KVM
  9e8745c7-52e3-4aae-984c-e198869ee2cc   pcacn002   Provisioned         KVM
  56a9ecda-2402-427f-92d1-7f9be57dba36   pcacn003   Provisioned         KVM

PCA-ADMIN> provisioningLock id=363a26f4-fa34-4e4c-8e17-a1671a0b77d1
PCA-ADMIN> maintenanceLock id=363a26f4-fa34-4e4c-8e17-a1671a0b77d1
```

3. Optionally, upgrade the server ILOM first.

   a. Enter the ILOM upgrade command.
   Syntax (entered on a single line):

   ```
   upgradeIlom
   hostIp=<ilom-ip>
   ```

   Example:

   ```
   PCA-ADMIN> upgradeIlom hostIp=100.96.0.64
   Data:
     Service request has been submitted. Upgrade Job Id = 1620921089806-
   ilom-21480 Upgrade Request Id = UWS-732d6fce-9f06-4329-b972-d093bee40010

   PCA-ADMIN> getUpgradeJob upgradeJobId=1620921089806-ilom-21480
   ```

   b. Wait 5 minutes to allow the ILOM upgrade job to complete. Then proceed to the host upgrade.

4. Enter the compute node upgrade command.

   Syntax (entered on a single line):

   ```
   upgradeCN
   hostIp=<compute-node-ip>
   [optional] imageLocation=<path-to-iso>
   [optional] isoChecksum=<iso-file-checksum>
   ```

   The parameters marked optional are deprecated in software version 3.0.2-b892153 and later. For earlier versions, include the ISO image parameters with the command.

Example:

```
PCA-ADMIN> upgradeCN hostIp=100.96.2.64 \
imageLocation="http://host.example.com/pca-<version>-<build>.iso" \
isoChecksum=240420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b
4c7f29026f0a5f58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7
Command: upgradeCN hostIp=100.96.2.64 imageLocation="http://host.example.com/pca-
<version>-<build>.iso"
isoChecksum=240420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b
4c7f29026f0a5f58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7
Status: Success
Time: 2021-09-26 06:35:38,884 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1630938939109-compute-7545
Upgrade Request Id = UWS-61736806-7e5a-4648-9259-07c54c39cacb
```

5. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id                                upgradeRequestId
commandName   result
  --                                ----------------
-----------   ------
  1630938939109-compute-7545        UWS-61736806-7e5a-4648-9259-07c54c39cacb
compute       Passed
  1632850650836-platform-68465      UWS-26dba234-9b52-426d-836c-ac11f37e717f
platform      Passed
  1632849609034-kubernetes-35545    UWS-edfa3b32-c32a-4b67-8df5-2357096052bf
kubernetes    Passed

PCA-ADMIN> getupgradejob upgradeJobId=1630938939109-compute-7545
Command: getupgradejob upgradeJobId=1630938939109-compute-7545
Status: Success
Time: 2021-09-26 08:15:03,208 UTC
Data:
  Upgrade Request Id = UWS-61736806-7e5a-4648-9259-07c54c39cacb
  Name = compute
  Start Time = 2021-09-26T06:35:39
  End Time = 2021-09-26T06:45:55
  Pid = 7545
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_compute_2021_09_26-06.35.39.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":"100.96.2.64","r
esult_override":null,"log_level":null,"switch_type":null,"precheck_status":false,"t
ask_time":0,"fail_halt":false,"fail_upgrade":null,"component_names":null,"upgrade_t
o":null,"image_location":null,"epld_image_location":null,"expected_iso_checksum":nu
ll,"checksum":null,"composition_id":null,"request_id":"UWS-61736806-7e5a-4648-9259-
07c54c39cacb","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 616
  Tasks 1 - Name = Copy Scripts
  Tasks 1 - Description = Copy scripts to shared storage
  Tasks 1 - Time = 2021-09-26T06:35:39
[...]
```

6. When the compute node upgrade has completed successfully and the node has rebooted, release the locks.

For more information, refer to the section "Performing Compute Node Operations". It can be found in the chapter Hardware Administration of the Oracle Private Cloud Appliance Administrator Guide.

```
PCA-ADMIN> maintenanceUnlock id=363a26f4-fa34-4e4c-8e17-a1671a0b77d1
PCA-ADMIN> provisioningUnlock id=363a26f4-fa34-4e4c-8e17-a1671a0b77d1
```

7. Proceed to the next compute node and repeat this procedure.

6

Performing a Full Management Node Upgrade

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

A full management node upgrade is a convenient way to upgrade all the required components on all three management nodes using just a single command. As part of this process, the following components are upgraded, in this specific order:

1. host operating system
2. Clustered MySQL database
3. secret service (including Etcd and Vault)
4. Kubernetes container orchestration packages
5. containerized microservices
6. Oracle Cloud Infrastructure images

> **Note:**
>
> To obtain the latest images on systems with software versions 3.0.2-b852928 and earlier, you must manually perform the steps described in Upgrading Oracle Cloud Infrastructure Images.

> **Note:**
>
> In software version 3.0.2-b892153 and later the Upgrader service uses the upgrade plan, generated during the pre-upgrade process, to determine which specific components need to be upgraded. If a component is already at the required version, it is skipped. A same-version upgrade can be forced using the Service Web UI or Service CLI command option, if necessary. For example: `upgradeKubernetes force=True`.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

The Create Request window appears. Choose *Upgrade* as the Request Type.

3. Select the appropriate upgrade request type.

   For a full management node upgrade, select Upgrade MN.

> **Caution:**
>
> Oracle strongly recommends using the verify-only option on the first run. This starts a verification job for each component. Before you start the actual upgrade, all verification jobs must be completed, which could take around 45 minutes in total.

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

4. Fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Image Location:** This parameter is deprecated.
   - **ISO Checksum:** This parameter is deprecated.
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

> **Note:**
>
> After upgrade, if the upgrade plan specifies that the management nodes must be rebooted for the changes to take effect, a reboot is performed as part of the upgrade process. No administrator action is required.
>
> If the appliance is running software version 3.0.2-b892153 or earlier, all management nodes are rebooted as part of the upgrade process.

Using the Service CLI

1. Run the upgrade command in verify-only mode.

> **Caution:**
>
> Oracle strongly recommends using the verify-only option (`verifyOnly=True`) on the first run. This starts a verification job for each component. Before you start the actual upgrade, all verification jobs must be completed, which could take around 45 minutes in total.

> **Note:**
>
> This function is available when the appliance is running software version 3.0.2-b892153 or later.

a. Start the upgrade request in verify-only mode.

```
PCA-ADMIN> upgradeFullMN verifyOnly=True
Data: Service request has been submitted. Upgrade Request Id =
UWS-9776b0fe-7f5f-4e46-9e3f-ceb4b1702056
```

b. Find the associated upgrade jobs and note the IDs.

```
PCA-ADMIN> getUpgradeJobs
Data:
  id
upgradeRequestId                            commandName         result
  --
----------------                            -----------         ------
  1698788568097-kubernetes_verify-32571   UWS-9776b0fe-7f5f-4e46-9e3f-
ceb4b1702056   kubernetes_verify   Passed
  1698788265717-vault_verify-27153        UWS-9776b0fe-7f5f-4e46-9e3f-
ceb4b1702056   vault_verify        Passed
  1698787959640-etcd_verify-26413         UWS-9776b0fe-7f5f-4e46-9e3f-
ceb4b1702056   etcd_verify         Passed
  1698787657287-mysql_verify-19656        UWS-9776b0fe-7f5f-4e46-9e3f-
ceb4b1702056   mysql_verify        Passed
  1698787353293-host_verify-18867         UWS-9776b0fe-7f5f-4e46-9e3f-
ceb4b1702056   host_verify         Passed
  1698787049308-host_verify-12573         UWS-9776b0fe-7f5f-4e46-9e3f-
ceb4b1702056   host_verify         Passed
  1698786925154-host_verify-36612         UWS-9776b0fe-7f5f-4e46-9e3f-
ceb4b1702056   host_verify         Passed
  1616912591633-prepare-67207             UWS-122d3628-1675-42fb-b581-
d3824caa329c   prepare             Passed
```

c. Check the status in the upgrade job details.

```
PCA-ADMIN> getUpgradeJob upgradeJobId=1698787353293-host_verify-18867
Data:
  Upgrade Request Id = UWS-9776b0fe-7f5f-4e46-9e3f-ceb4b1702056
  Name = host_verify
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_host_os_pcamn03_verify_2023_06_16-13.07.11.log
  Status = Passed
```

2. When the upgrade verification has completed successfully, start the actual full management node upgrade.

```
PCA-ADMIN> upgradeFullMN
Data:
  Service request has been submitted. Upgrade Request Id =
UWS-39329657-1051-4267-8c5a-9314f8e63a64
```

3. Use the request ID to check the status of the upgrade process.

   As the full management node upgrade is a multi-component upgrade process, there are multiple upgrade jobs associated with the upgrade request. You can filter for those jobs based on the request ID. Using the job ID, you can drill down into the details of each upgrade job.

```
PCA-ADMIN> getUpgradeJobs requestId=UWS-39329657-1051-4267-8c5a-9314f8e63a64
Data:
  id
upgradeRequestId                              commandName   result
  --
----------------                              -----------   ------
  1634579161548-oci-47486
UWS-39329657-1051-4267-8c5a-9314f8e63a64   oci           Passed
  1634578760906-platform-66082
UWS-39329657-1051-4267-8c5a-9314f8e63a64   platform      Passed
  1634578263434-kubernetes-63574
UWS-39329657-1051-4267-8c5a-9314f8e63a64   kubernetes    Passed
  1634578012353-vault-51696
UWS-39329657-1051-4267-8c5a-9314f8e63a64   vault         Passed
  1634577380954-etcd-46337
UWS-39329657-1051-4267-8c5a-9314f8e63a64   etcd          Passed
  1634577341291-mysql-40127
UWS-39329657-1051-4267-8c5a-9314f8e63a64   mysql         Passed
  1634576985926-host-36556
UWS-39329657-1051-4267-8c5a-9314f8e63a64   host          Passed
  1634576652071-host-27088
UWS-39329657-1051-4267-8c5a-9314f8e63a64   host          Passed
  1634576191050-host-24909
UWS-39329657-1051-4267-8c5a-9314f8e63a64   host          Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1634576652071-host-27088
Data:
  Upgrade Request Id = UWS-39329657-1051-4267-8c5a-9314f8e63a64
  Composition Id = 1
  Name = host
  Start Time = 2023-05-24T07:04:12
  End Time = 2023-05-24T07:05:22
  Pid = 27088
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_host_os_2023_05_24-07.04.12.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":"100.96.2.
35","result_override":null,"log_level":null,"switch_type":null,"precheck_stat
us":false,"task_time":0,"fail_halt":false,"fail_upgrade":null,"component_name
s":null,"upgrade_to":null,"image_location":"file:///nfs/shared_storage/
pca-3.0.1-
b544818.iso","epld_image_location":null,"expected_iso_checksum":null,"checksu
m":"240420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b4c
7f29026f0a5f58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7","composition_id":"1"
,"request_id":"UWS-39329657-1051-4267-8c5a-9314f8e63a64","display_task_plan":
false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 139
  Tasks 1 - Name = Validate Image Location
  Tasks 1 - Description = Verify that the image exists at the specified
location and is correctly named
  Tasks 1 - Time = 2023-05-24T17:04:16
[...]
```

The output of the `getUpgradeJob` command provides detailed information about the tasks performed during the upgrade procedure. It displays descriptions, time stamps, duration, and success or failure. Whenever an upgrade operation fails, the command output indicates which task has failed. For in-depth troubleshooting you

can search the log file at the location provided near the start of the command output.

> **Note:**
>
> After upgrade, if the upgrade plan specifies that the management nodes must be rebooted for the changes to take effect, a reboot is performed as part of the upgrade process. No administrator action is required.
>
> If the appliance is running software version 3.0.2-b892153 or earlier, all management nodes are rebooted as part of the upgrade process.

7

Upgrading Individual Components

The granular upgrade mechanism allows you to perform upgrade procedures for individual hardware and software components. Besides the components included in the management node upgrade, you can also upgrade different categories of firmware, and the operating system and appliance-specific software on the compute nodes.

Individual component upgrades must also comply with the prescribed order of operations. A component can only be upgraded on condition that the preceding components in the list are already at the required version. For more information, see Checking Upgrade Plan Status and Progress.

> **Note:**
>
> In software version 3.0.2-b892153 and later the Upgrader service uses the upgrade plan, generated during the pre-upgrade process, to determine whether a component needs to be upgraded. If a component is already at the required version, the upgrade command does start an upgrade job, but it is completed immediately because the upgrade plan indicates there is nothing to do.
>
> Practically speaking, when a component is already at the required version, the upgrade procedure is skipped. However, a same-version upgrade can be forced using the Service Web UI or Service CLI command option, if necessary. For example: `upgradeKubernetes force=True`.

Upgrading the Management Node Operating System with Appliance Software 3.0.2-b852928 or Earlier

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

The Oracle Linux host operating system of the management nodes must be upgraded one node at a time; a rolling upgrade of all management nodes is not possible. This upgrade process, which involves updating the kernel and system packages, must always be initiated from the management node that holds the cluster virtual IP. Thus, in a three-management-node cluster, when you have upgraded two management nodes, you must reassign the cluster virtual IP to one of the upgraded management nodes and execute the final upgrade command from that node.

You must upgrade management nodes one at a time, using each one's internal IP address as a command parameter. To obtain the host IP addresses, use the Service CLI command `show ManagementNode name=<node_name>` and look for the `Ip Address` in the output.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: Upgrade Host.
4. Fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Host IP:** Enter the management node's assigned IP address in the internal administration network. This is an IP address in the internal 100.96.2.0/23 range.
   - **Image Location:** This parameter is deprecated.
   - **ISO Checksum:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

> **Note:**
>
> After upgrade, the management nodes must all be rebooted for the changes to take effect. However, this step is part of the upgrade process, so no administrator action is required.

Using the Service CLI

1. Get the IP address of the management node for which you intend to upgrade the host operating system.
2. Run the Service CLI from the management node that holds the management cluster virtual IP.
   a. Log on to one of the management nodes and check the status of the cluster.

```
# ssh root@pcamn01
# pcs status
Cluster name: mncluster
Stack: corosync
Current DC: pcamn02 (version 1.1.23-1.0.1.el7-9acf116022) - partition
with quorum

Online: [ pcamn01 pcamn02 pcamn03 ]

Full list of resources:
```

```
 scsi_fencing         (stonith:fence_scsi):          Stopped (disabled)
 Resource Group: mgmt-rg
     vip-mgmt-int     (ocf::heartbeat:IPaddr2):      Started    pcamn02
     vip-mgmt-host    (ocf::heartbeat:IPaddr2):      Started    pcamn02
     vip-mgmt-ilom    (ocf::heartbeat:IPaddr2):      Started    pcamn02
     vip-mgmt-lb      (ocf::heartbeat:IPaddr2):      Started    pcamn02
     vip-mgmt-ext     (ocf::heartbeat:IPaddr2):      Started    pcamn02
     l1api            (systemd:l1api):               Started    pcamn02
     haproxy          (ocf::heartbeat:haproxy):      Started    pcamn02
     pca-node-state   (systemd:pca_node_state):      Started    pcamn02
     dhcp             (ocf::heartbeat:dhcpd):        Started    pcamn02
     hw-monitor       (systemd:hw_monitor):          Started    pcamn02
     healthcheck      (systemd:healthcheck):         Started    pcamn02
     rabbitmq-monitor (systemd:rabbitmq_monitor):    Started    pcamn02

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

In this example, the command output indicates that the node with host name `pcamn02` currently holds the cluster virtual IP.

b. Log in to the management node with the virtual IP and launch the Service CLI.

```
# ssh pcamn02
# ssh admin@localhost -p 30006
PCA-ADMIN>
```

3. Enter the upgrade command.

Syntax (entered on a single line):

```
upgradeHost
hostIp=<management-node-ip>
```

Example:

```
PCA-ADMIN> upgradeHost hostIp=100.96.2.35
Command: upgradeHost hostIp=100.96.2.35
Status: Success
Time: 2021-09-25 05:47:02,735 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1632990827394-host-56156
Upgrade Request Id = UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
```

4. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id                              upgradeRequestId
commandName   result
  --                              ----------------
-----------   ------
  1632990827394-host-56156        UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
host          Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1632990827394-host-56156
Command: getUpgradeJob upgradeJobId=1632990827394-host-56156
Status: Success
Time: 2021-09-25 05:54:28,054 UTC
Data:
  Upgrade Request Id = UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
```

```
  Composition Id = 1
  Name = host
  Start Time = 2021-09-25T05:47:02
  End Time = 2021-09-25T05:48:38
  Pid = 56156
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_host_os_2021_09_25-05.47.02.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":"100.96.2.
35","result_override":null,"log_level":null,"switch_type":null,"precheck_stat
us":false,"task_time":0,"fail_halt":false,"fail_upgrade":null,"component_name
s":null,"upgrade_to":null,"image_location":"http://host.example.com/
pca-3.0.1-
b535176.iso","epld_image_location":null,"expected_iso_checksum":null,"checksu
m":"240420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b4c
7f29026f0a5f58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7","composition_id":"1"
,"request_id":"UWS-1a97a8d9-54ef-478d-
a0c0-348a17ba6755","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 96
  Tasks 1 - Name = Validate Image Location
  Tasks 1 - Description = Verify that the image exists at the specified
location and is correctly named
  Tasks 1 - Time = 2021-09-25T05:47:02
  Tasks 2 - Name = Validate Image Location
[...]
```

5. When the first management node host operating system upgrade has completed successfully, execute the same command for the next management node.

```
PCA-ADMIN> upgradeHost hostIp=100.96.2.33
```

6. When the second management node host operating system upgrade has completed successfully, exit the Service CLI and move the cluster virtual IP to one of the upgraded nodes.

```
PCA-ADMIN> exit
Connection to localhost closed.
# pcs resource move mgmt-rg pcamn01
# pcs status
Cluster name: mncluster
Stack: corosync
[...]
 scsi_fencing   (stonith:fence_scsi):   Stopped (disabled)
 Resource Group: mgmt-rg
     vip-mgmt-int       (ocf::heartbeat:IPaddr2):       Started pcamn01
     vip-mgmt-host      (ocf::heartbeat:IPaddr2):       Started pcamn01
[...]
```

   Moving the cluster virtual IP to another management node should only take a number of seconds.

7. Log in to the management node with the virtual IP and launch the Service CLI to execute the host operating system upgrade for the final management node.

```
# ssh pcamn01
# ssh admin@localhost -p 30006
PCA-ADMIN> upgradeHost hostIp=100.96.2.34
```

   When this upgrade has completed successfully, the operating system on all management nodes is up-to-date.

> **Note:**
>
> After upgrade, the management nodes must all be rebooted for the changes to take effect. However, this step is part of the upgrade process, so no administrator action is required.

Upgrading the Management Node Operating System

> **Caution:**
>
> Follow this procedure only when the appliance is running software version 3.0.2-b892153 or later. Otherwise, additional steps are required to move the primary role in the cluster between node upgrades. Follow this procedure instead: Upgrading the Management Node Operating System with Appliance Software 3.0.2-b852928 or Earlier

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

The Oracle Linux host operating system of the management nodes must be upgraded one node at a time; a rolling upgrade of all management nodes is not possible. This upgrade process, which involves updating the kernel and system packages, detects which of the nodes in the three-management-node cluster owns the virtual IP and the primary role. When the current primary node is upgraded, the primary role is first transferred to another node in the cluster. This configuration change occurs in the background and requires no separate or additional intervention from an administrator.

> **Note:**
>
> In case the ILOM also needs to be upgraded, you can integrate it into this procedure by executing the optional steps. The combined procedure eliminates the need to evacuate and reboot the same node twice.

You must upgrade management nodes one at a time, using each one's internal IP address as a command parameter. To obtain the host IP addresses, use one of these Service CLI commands:

- Enter `show ManagementNode name=<node_name>` and look for the `Ip Address` in the output.

```
PCA-ADMIN> show ManagementNode name=pcamn01
[...]
Data:
```

```
Id = d9f10197-9a7a-4602-8440-f5f43f573f65
Type = ManagementNode
HW Id = AK0MYPCA3X
MAC Address = a8:69:8c:05:e0:d8
Ip Address = 100.96.2.33
ILOM Ip Address = 100.96.0.33
ILOM MAC Address = A8:69:8C:05:E0:DB
[...]
```

- On systems running software version 3.0.2-b892153 or later, enter `getServerIP hostName=<node_name>`.

```
PCA-ADMIN> getServerIP hostName=pcamn01
[...]
Data:
  status = success
  data = 100.96.2.33
```

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. Optionally, upgrade the server ILOM first.
   a. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch. The Create Request window appears.
   b. Choose *Upgrade* as the Request Type. Select the appropriate upgrade request type: Upgrade ILOM.

      Fill out the server's assigned IP address in the ILOM network. This is an IP address in the internal 100.96.0.0/23 range.
   c. Click Create Request. The new upgrade request appears in the Upgrade Jobs table.
   d. Wait 5 minutes to allow the ILOM upgrade job to complete. Then proceed to the host upgrade.
3. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
4. Select the appropriate upgrade request type: Upgrade Host.
5. Fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Host IP:** Enter the management node's assigned IP address in the internal administration network. This is an IP address in the internal 100.96.2.0/23 range.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
6. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

> **Note:**
>
> After upgrade, if the upgrade plan specifies that the management nodes must be rebooted for the changes to take effect, a reboot is performed as part of the upgrade process. No administrator action is required.
>
> If the appliance is running software version 3.0.2-b892153 or earlier, all management nodes are rebooted as part of the upgrade process.

Using the Service CLI

1. Get the IP address of the management node for which you intend to upgrade the host operating system.
2. Optionally, upgrade the server ILOM first.
   a. Enter the ILOM upgrade command.
   Syntax (entered on a single line):

   ```
   upgradeIlom
   hostIp=<ilom-ip>
   ```

   Example:

   ```
   PCA-ADMIN> upgradeIlom hostIp=100.96.0.35
   Data:
     Service request has been submitted. Upgrade Job Id = 1632990827394-
   ilom-23871 Upgrade Request Id = UWS-1a97a8d9-9f06-478d-a0c0-d093bee4672

   PCA-ADMIN> getUpgradeJob upgradeJobId=1632990827394-ilom-23871
   ```

   b. Wait 5 minutes to allow the ILOM upgrade job to complete. Then proceed to the host upgrade.
3. Enter the management node host upgrade command.

   Syntax (entered on a single line):

   ```
   upgradeHost
   hostIp=<management-node-ip>
   ```

   Example:

   ```
   PCA-ADMIN> upgradeHost hostIp=100.96.2.35
   Command: upgradeHost hostIp=100.96.2.35
   Status: Success
   Time: 2021-09-25 05:47:02,735 UTC
   Data:
     Service request has been submitted. Upgrade Job Id = 1632990827394-host-56156
   Upgrade Request Id = UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
   ```

4. Use the request ID and the job ID to check the status of the upgrade process.

   ```
   PCA-ADMIN> getUpgradeJobs
     id                              upgradeRequestId
   commandName    result
     --                              ----------------
   -----------    ------
     1632990827394-host-56156        UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
   host           Passed

   PCA-ADMIN> getUpgradeJob upgradeJobId=1632990827394-host-56156
   ```

```
Command: getUpgradeJob upgradeJobId=1632990827394-host-56156
Status: Success
Time: 2021-09-25 05:54:28,054 UTC
Data:
  Upgrade Request Id = UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
  Composition Id = 1
  Name = host
  Start Time = 2021-09-25T05:47:02
  End Time = 2021-09-25T05:48:38
  Pid = 56156
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_host_os_2021_09_25-05.47.02.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":"100.96.2.
35","result_override":null,"log_level":null,"switch_type":null,"precheck_stat
us":false,"task_time":0,"fail_halt":false,"fail_upgrade":null,"component_name
s":null,"upgrade_to":null,"image_location":"http://host.example.com/
pca-3.0.1-
b535176.iso","epld_image_location":null,"expected_iso_checksum":null,"checksu
m":"240420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b4c
7f29026f0a5f58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7","composition_id":"1"
,"request_id":"UWS-1a97a8d9-54ef-478d-
a0c0-348a17ba6755","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 96
  Tasks 1 - Name = Validate Image Location
  Tasks 1 - Description = Verify that the image exists at the specified
location and is correctly named
  Tasks 1 - Time = 2021-09-25T05:47:02
  Tasks 2 - Name = Validate Image Location
[...]
```

5. When the first management node host operating system upgrade has completed successfully, execute the same command for the second management node. When that upgrade has completed successfully, execute the same command for the third management node.

```
PCA-ADMIN> upgradeHost hostIp=100.96.2.33
PCA-ADMIN> upgradeHost hostIp=100.96.2.34
```

When all three operating system upgrades have completed successfully, the management node cluster is up-to-date.

> **Note:**
>
> After upgrade, if the upgrade plan specifies that the management nodes must be rebooted for the changes to take effect, a reboot is performed as part of the upgrade process. No administrator action is required.
>
> If the appliance is running software version 3.0.2-b892153 or earlier, all management nodes are rebooted as part of the upgrade process.

Upgrading the MySQL Cluster Database

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

The MySQL Cluster database is upgraded independently of the management node host operating system; the MySQL packages are deliberately kept separate from the Oracle Linux upgrade.

The MySQL Cluster database upgrade is a rolling upgrade: with one command the upgrade is executed on each of the three management nodes.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: Upgrade MySQL.
4. If required, fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Image Location:** This parameter is deprecated.
   - **ISO Checksum:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

Using the Service CLI

1. Enter the upgrade command.

```
PCA-ADMIN> upgradeMySQL
Command: upgradeMySQL
Status: Success
Time: 2021-09-25 09:21:16,264 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1632995409822-mysql-83013
Upgrade Request Id = UWS-77bc0c30-7ff5-4c50-ad09-6f96907e22e1
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id                                upgradeRequestId
commandName   result
  --                                ----------------
-----------   ------
```

```
  1632995409822-mysql-83013        UWS-77bc0c30-7ff5-4c50-
ad09-6f96907e22e1   mysql         Passed
  1632926926773-host-32993         UWS-fef3b663-45b7-4177-
a041-26f73e68848d   host          Passed
  1632990827394-host-56156         UWS-1a97a8d9-54ef-478d-
a0c0-348a17ba6755   host          Passed
  1632990493570-host-6646          UWS-4c78f3ef-ac42-4f32-9483-
bb43a309faa3   host          Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1632995409822-mysql-83013
Command: getUpgradeJob upgradeJobId=1632995409822-mysql-83013
Status: Success
Time: 2021-09-25 09:24:27,874 UTC
Data:
  Upgrade Request Id = UWS-77bc0c30-7ff5-4c50-ad09-6f96907e22e1
  Name = mysql
  Start Time = 2021-09-25T09:21:16
  End Time = 2021-09-25T09:22:04
  Pid = 83013
  Host = pcamn01
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_mysql_cluster_2021_09_25-09.21.16.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":null,"resu
lt_override":null,"log_level":null,"switch_type":null,"precheck_status":false
,"task_time":0,"fail_halt":false,"fail_upgrade":null,"component_names":null,"
upgrade_to":null,"image_location":"http://host.example.com/pca-3.0.1-
b535176.iso","epld_image_location":null,"expected_iso_checksum":null,"checksu
m":"240420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b4c
7f29026f0a5f58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7","composition_id":nul
l,"request_id":"UWS-77bc0c30-7ff5-4c50-
ad09-6f96907e22e1","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 48
  Tasks 1 - Name = Validate Image Location
  Tasks 1 - Description = Verify that the image exists at the specified
location and is correctly named
  Tasks 1 - Time = 2021-09-25T09:21:16
[...]
```

Upgrading the Secret Service

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

The secret service contains two components that need to be upgraded separately in this particular order: first Etcd, then Vault.

The Etcd and Vault upgrades are rolling upgrades: each upgrade is executed on all three management nodes with one command.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.

2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.

3. Select the appropriate upgrade request type: Upgrade Etcd.

4. If required, fill out the upgrade request parameters:

   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Image Location:** This parameter is deprecated.
   - **ISO Checksum:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".

5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

6. When the Etcd upgrade has completed successfully, repeat this procedure to create an upgrade request for Vault.

Using the Service CLI

1. Enter the two upgrade commands. Wait until the Etcd upgrade is finished before starting the Vault upgrade.

```
PCA-ADMIN> upgradeEtcd
Command: upgradeEtcd
Status: Success
Time: 2021-09-25 10:24:52,177 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1632826770954-etcd-26973
Upgrade Request Id = UWS-fec15d32-fc2b-48bd-9ae0-62f49587a284

PCA-ADMIN> upgradeVault
Command: upgradeVault
Status: Success
Time: 2021-09-25 10:38:25,417 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1632850933353-vault-16966
Upgrade Request Id = UWS-352df3d1-c21f-441b-8f6e-9381ac075906
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id                              upgradeRequestId
commandName   result
  --                              ----------------
-----------   ------
  1632995409822-mysql-83013       UWS-77bc0c30-7ff5-4c50-ad09-6f96907e22e1
mysql         Passed
  1632850933353-vault-16966       UWS-352df3d1-c21f-441b-8f6e-9381ac075906
vault         Passed
  1632826770954-etcd-26973        UWS-fec15d32-fc2b-48bd-9ae0-62f49587a284
etcd          Passed
  1632926926773-host-32993        UWS-fef3b663-45b7-4177-a041-26f73e68848d
host          Passed
  1632990827394-host-56156        UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
host          Passed
  1632990493570-host-6646         UWS-4c78f3ef-ac42-4f32-9483-bb43a309faa3
host          Passed
```

```
PCA-ADMIN> getUpgradeJob upgradeJobId=1632850933353-vault-16966
Command: getUpgradeJob upgradeJobId=1632850933353-vault-16966
Status: Success
Time: 2021-09-25 10:39:31,308 UTC
Data:
  Upgrade Request Id = UWS-352df3d1-c21f-441b-8f6e-9381ac075906
  Name = vault
  Start Time = 2021-09-25T10:38:25
  End Time = 2021-09-25T10:39:07
  Pid = 16966
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_vault_2021_09_25-10.38.25.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":null,"resu
lt_override":null,"log_level":null,"switch_type":null,"precheck_status":false
,"task_time":0,"fail_halt":false,"fail_upgrade":null,"component_names":null,"
upgrade_to":null,"image_location":"http://host.example.com/pca-3.0.1-
b535176.iso","epld_image_location":null,"expected_iso_checksum":null,"checksu
m":"240420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b4c
7f29026f0a5f58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7","composition_id":nul
l,"request_id":"UWS-352df3d1-
c21f-441b-8f6e-9381ac075906","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 42
  Tasks 1 - Name = Check Vault Running Status
  Tasks 1 - Description = Check vault service running status is healthy
  Tasks 1 - Time = 2021-09-25T10:38:25
[...]
```

Upgrading the Kubernetes Cluster

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

The Kubernetes container orchestration environment upgrade is also kept separate from the operating system. With a single command, all Kubernetes packages, such as kubeadm, kubectl and kubelet, are upgraded on the three management nodes and all the compute nodes. Note that this upgrade does not include the microservices running within the Kubernetes cluster.

For dependency reasons, Kubernetes must be upgraded after the management node host operating system. The Kubernetes upgrade command has no mandatory parameters.

About the Kubernetes Upgrade Process

To ensure compatibility and continuation of service, Kubernetes must be upgraded one version at a time. Skipping versions – major or minor – is not supported. The Private Cloud Appliance Upgrader manages this process by upgrading or patching all parts of the Kubernetes cluster to the next available version, repeating the same sequence of

operations until the entire environment runs the latest Kubernetes version available from the appliance software repositories.

Upgrading or patching the Kubernetes cluster is a time-consuming process that involves the Private Cloud Appliancemanagement nodes and compute nodes. Each additional compute node extends the process by appoximately 10 minutes for each incremental version of Kubernetes.

With appliance software version 3.0.2-b925538, the container orchestration environment is upgraded or patched from Kubernetes version 1.20.x to version 1.25.y, meaning the entire process must run 5 times. After each successful run, the repository is synchronized to retrieve the next required version. However, with this version of the appliance software the repository is reconfigured to allow multiple versions of the Kubernetes packages, so the resync will no longer be required.

Each individual Kubernetes node upgrade is expected to take around 10 minutes. Testing indicates that upgrading or patching the Private Cloud Appliance Kubernetes cluster from version 1.20 to version 1.25 takes approximately 4-5 hours for a base rack configuration with 3 management nodes and 3 compute nodes. On a full rack with 20 compute nodes the entire process requires at least 9 hours and may take up to 18 hours to complete. The estimated time for the rack's specific configuration is reported in the upgrade plan.

To monitor the upgrade or patching progress, periodically check the job status or the logs.

- Check job status through the Service CLI: `getUpgradeJob upgradeJobId=`***`<id>`***
- View Upgrader logs on a management node: `tail -f /nfs/shared_storage/pca_upgrader/log/pca-upgrader_kubernetes_cluster_`***`<time_stamp>`***`.log`.

During Kubernetes upgrade or patching, certain services could be temporarily unavailable.

- The Compute Web UI, Service Web UI, OCI CLI, and Service CLI can all become temporarily unavailable. Users should wait a few minutes before attempting their operations again. Administrative operations in the Service Enclave (UI or CLI) must be avoided during upgrade or patching.
- When the Kubernetes upgrade is initiated, the Kubernetes Workload Monitoring Operator (Sauron service) is taken down. As a result, the Grafana, Prometheus, and other Sauron ingress endpoints cannot be accessed. They become available again after both the Kubernetes cluster and the containerized microservices (platform layer) upgrade or patching processes have been completed.

Managing Unprovisioned Compute Nodes

If you upgrade or patch the Kubernetes cluster on a Private Cloud Appliance that contains unprovisioned compute nodes, there could be provisioning issues later. Because those compute nodes were not part of the Kubernetes cluster when the newer version was applied, you may need to rediscover them first.

If compute node provisioning fails after upgrading or patching the Kubernetes cluster, log on to one of the management nodes using ssh. Rediscover the unprovisioned compute nodes by running the following command with the appropriate host names:

```
# pca-admin compute node rediscover --hostname pcacn000
```

When the compute nodes have been rediscovered, provisioning is expected to work as intended.

For more information about provisioning, refer to "Performing Compute Node Operations" in the chapter Hardware Administration of the Oracle Private Cloud Appliance Administrator Guide.

Upgrade the Kubernetes Cluster Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: Upgrade Kubernetes.
4. If required, fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Image Location:** This parameter is deprecated.
   - **ISO Checksum:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

Upgrade the Kubernetes Cluster Using the Service CLI

1. Enter the upgrade command.

```
PCA-ADMIN> upgradeKubernetes
Command: upgradeKubernetes
Status: Success
Time: 2021-09-26 17:20:09,423 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1632849609034-
kubernetes-35545 Upgrade Request Id = UWS-edfa3b32-
c32a-4b67-8df5-2357096052bf
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id
upgradeRequestId                          commandName   result
  --
----------------                          -----------   ------
  1632849609034-kubernetes-35545   UWS-edfa3b32-
c32a-4b67-8df5-2357096052bf   kubernetes    Passed
  1632826770954-etcd-26973         UWS-fec15d32-
fc2b-48bd-9ae0-62f49587a284   etcd          Passed
  1632850933353-vault-16966        UWS-352df3d1-
c21f-441b-8f6e-9381ac075906   vault         Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1632849609034-kubernetes-35545
Command: getUpgradeJob upgradeJobId=1632849609034-kubernetes-35545
Status: Success
Time: 2021-09-26 17:43:38,443 UTC
Data:
  Upgrade Request Id = UWS-edfa3b32-c32a-4b67-8df5-2357096052bf
  Name = kubernetes
  Start Time = 2021-09-26T17:20:09
```

```
  End Time = 2021-09-26T17:21:52
  Pid = 35545
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_kubernetes_cluster_2021_09_26-17.20.09.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":null,"result_ove
rride":null,"log_level":null,"switch_type":null,"precheck_status":false,"task_time"
:0,"fail_halt":false,"fail_upgrade":null,"component_names":null,"upgrade_to":null,"
image_location":"http://host.example.com/pca-3.0.1-
b535176.iso","epld_image_location":null,"expected_iso_checksum":null,"checksum":"24
0420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b4c7f29026f0a5f
58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7","composition_id":null,"request_id":"UW
S-edfa3b32-
c32a-4b67-8df5-2357096052bf","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 249
  Tasks 1 - Name = Retrieving Cluster Status
  Tasks 1 - Description = Retrieving cluster status and upgrade data from the
kubernetes nodes
  Tasks 1 - Time = 2021-09-26T17:20:10
[...]
```

Upgrading the Microservices

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

The microservices upgrade covers both the internal services of the platform layer, and the administrative and user-level services exposed through the infrastructure services layer.

> **Note:**
>
> In specific circumstances it is possible to upgrade certain platform services individually, by adding an optional JSON string to the command. This option should not be used unless Oracle provides explicit instructions to do so.

The containerized microservices have their own separate upgrade mechanism. A service is upgraded if a new Helm deployment chart and container image are found in the ISO image. When a new deployment chart is detected during the upgrade process, the pods running the services are restarted with the new container image.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: Upgrade Platform.

4. If required, fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Image Location:** This parameter is deprecated.
   - **ISO Checksum:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

Using the Service CLI

1. Enter the upgrade command.

```
PCA-ADMIN> upgradePlatform
Command: upgradePlatform
Status: Success
Time: 2021-09-26 20:48:41,452 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1632850650836-
platform-68465 Upgrade Request Id = UWS-26dba234-9b52-426d-836c-ac11f37e717f
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id
upgradeRequestId                          commandName   result
  --
----------------                          -----------   ------
  1632850650836-platform-68465     UWS-26dba234-9b52-426d-836c-
ac11f37e717f   platform      Passed
  1632849609034-kubernetes-35545   UWS-edfa3b32-
c32a-4b67-8df5-2357096052bf   kubernetes    Passed
  1632826770954-etcd-26973         UWS-fec15d32-
fc2b-48bd-9ae0-62f49587a284   etcd          Passed
  1632850933353-vault-16966        UWS-352df3d1-
c21f-441b-8f6e-9381ac075906   vault         Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1632850650836-platform-68465
Command: getUpgradeJob 1632850650836-platform-68465
Status: Success
Time: 2021-09-26 21:03:16,264 UTC
Data:
  Upgrade Request Id = UWS-26dba234-9b52-426d-836c-ac11f37e717f
  Name = kubernetes
  Start Time = 2021-09-26T20:48:41
  End Time = 2021-09-26T20:59:34
  Pid = 68465
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_platform_services_2021_09_26-20.48.41.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":null,"resu
lt_override":null,"log_level":null,"switch_type":null,"precheck_status":false
,"task_time":0,"fail_halt":false,"fail_upgrade":null,"component_names":null,"
upgrade_to":null,"image_location":"http://host.example.com/pca-3.0.1-
b535176.iso","epld_image_location":null,"expected_iso_checksum":null,"checksu
m":"240420cfb9478f6fd026f0a5fa0e998e086275fc45e207fb5631e2e99732e192e8e9d1b4c
```

```
7f29026f0a5f58dadc4d792d0cfb0279962838e95a0f0a5fa31dca7","composition_id":null,"req
uest_id":"UWS-26dba234-9b52-426d-836c-
ac11f37e717f","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 653
  Tasks 1 - Name = Check All Ingress Endpoints
  Tasks 1 - Description = Check whether all ingress endpoints are up and running
  Tasks 1 - Time = 2021-09-26T20:48:42
[...]
```

Upgrading Oracle Cloud Infrastructure Images

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

When new Oracle Cloud Infrastructure Images become available and supported for Oracle Private Cloud Appliance, you can make them available for use in all existing tenancies with a single upgrade command. The images are stored in the `/nfs/shared_storage/oci_compute_images` directory on the ZFS Storage Appliance.

If you perform a full management node upgrade, the new images are automatically added to your environment, in which case you can skip this procedure. The image versions are tracked through the upgrade plan. Review the upgrade plan to verify if the images need to be upgraded.

An upgrade adds new Oracle Cloud Infrastructure Images to your environment, but it never removes any existing images. If you no longer need an image, you have the option to delete it using the `deletePlatformImage` command.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate patch request type: Upgrade OCIImages.
4. If required, fill out the request parameters:
   - **Advanced Options JSON:** Not available.
   - **Image Location:** This parameter is deprecated.
   - **ISO Checksum:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

Using the Service CLI

1. Enter the upgrade command.

```
PCA-ADMIN> upgradeOCIImages
Command: upgradeOCIImages
Status: Success
Time: 2023-01-18 19:48:41,452 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1680260388058-
oci-38288 Upgrade Request Id = UWS-e601b085-d59d-4ae4-82a3-a09b446686ff
```

2. Use the request ID and the job ID to check the status of the patching process.

```
PCA-ADMIN> getUpgradeJobs
Command: getUpgradeJobs
Status: Success
Time: 2023-01-18 19:58:34,745 UTC
Data:
  id                                upgradeRequestId
commandName   result
  --                                ----------------
-----------   ------
  1680260388058-oci-38288         UWS-e601b085-d59d-4ae4-82a3-a09b446686ff
oci           Passed
  1641839285475-kubernetes-94665  UWS-778b08bc-f579-492b-993d-915dcf581374
kubernetes    Passed
  1641838937541-platform-56313    UWS-bc4372ae-8f51-4b40-9306-992fb6459878
platform      Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1680260388058-oci-38288
Command: getUpgradeJob upgradeJobId=1680260388058-oci-38288
Status: Success
Time: 2023-01-18 20:00:43,804 UTC
Data:
  Upgrade Request Id = UWS-e601b085-d59d-4ae4-82a3-a09b446686ff
  Name = oci
[...]
```

Upgrading Firmware

> **Caution:**
>
> Ensure that all preparation steps for system upgrade have been completed. For instructions, see Preparing the Upgrade Environment.

Firmware is included in the ISO image for all component ILOMs, for the ZFS Storage Appliance, and for the switches. Select the instructions below for the component type you want to upgrade.

If you have completed the full management node upgrade to software version 3.0.2-b892153 or later, an upgrade plan has been generated. You can use it to check the progress of the appliance upgrade and determine which firmware upgrades still need to be performed.

Upgrading ILOMs

> **Note:**
>
> In case a server node needs both a host and ILOM upgrade, you can avoid having to reboot the same node twice by combining the two upgrades. Instructions are provided in the following sections:
>
> - Upgrading a Compute Node
> - Upgrading the Management Node Operating System

ILOM upgrades can be applied to management nodes and compute nodes; the firmware packages might be different per component type. You must upgrade ILOMs one at a time, using each one's internal IP address as a command parameter.

Before You Begin

To obtain the ILOM IP addresses, use the Service CLI commands shown below. You can list all ILOM IPs in the system or find the IP that corresponds with a specific machine.

```
PCA-ADMIN> getCNIloms
Data:
  status = success
  data 1 = 100.96.0.66
  data 2 = 100.96.0.64
  data 3 = 100.96.0.65

PCA-ADMIN> getMNIloms
Data:
  status = success
  data 1 = 100.96.0.33
  data 2 = 100.96.0.34
  data 3 = 100.96.0.35

PCA-ADMIN> getServerILOMIP hostName=pcacn002
Data:
  status = success
  data = 100.96.0.65

PCA-ADMIN> getServerILOMIP hostName=pcamn03
Data:
  status = success
  data = 100.96.0.35
```

Caution:

You must NOT upgrade the ILOM of the management node that holds the management virtual IP address, and thus the primary role in the cluster. To determine which management node has the primary role in the cluster, and make another node the primary, use the following Service CLI commands:

```
PCA-ADMIN> getPrimaryMgmtNode
  status = success
  data = pcamn01

PCA-ADMIN> updatePrimaryNode node=pcamn02
Data:
  status = success
  message = Successfully issued update primary node command

PCA-ADMIN> getPrimaryMgmtNode
  status = success
  data = pcamn02
```

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: Upgrade ILOM.
4. Fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Host IP:** Enter the component's assigned IP address in the ILOM network. This is an IP address in the internal 100.96.0.0/23 range.
   - **Image Location:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

Using the Service CLI

1. Get the IP address of the ILOM for which you intend to upgrade the firmware.
2. Enter the upgrade command.

   Syntax (entered on a single line):

   ```
   upgradeIlom
   hostIp=<ilom-ip>
   ```

   Example:

   ```
   PCA-ADMIN> upgradeIlom hostIp=100.96.0.66
   Command: upgradeIlom hostIp=100.96.0.66
   ```

```
Status: Success
Time: 2021-09-24 11:18:31,044 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1620921089806-ilom-21480
Upgrade Request Id = UWS-732d6fce-9f06-4329-b972-d093bee40010
```

3. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id                              upgradeRequestId
commandName   result
  --                              ----------------
-----------   ------
  1620921089806-ilom-21480        UWS-732d6fce-9f06-4329-b972-d093bee40010
ilom          Passed
  1632926926773-host-32993        UWS-fef3b663-45b7-4177-a041-26f73e68848d
host          Passed
  1632990827394-host-56156        UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
host          Passed
  1632990493570-host-6646         UWS-4c78f3ef-ac42-4f32-9483-bb43a309faa3
host          Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1620921089806-ilom-21480
Command: getUpgradeJob 1620921089806-ilom-21480
Status: Success
Time: 2021-09-24 11:24:49,243 UTC
Data:
  Upgrade Request Id = UWS-732d6fce-9f06-4329-b972-d093bee40010
  Name = ilom
  Start Time = 2021-09-24 11:18:32
  End Time = 2021-09-24 11:21:18
  Pid = 21480
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_ilom_firmware_2021_09_24-11.18.31.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":"100.96.0.66","r
esult_override":null,"log_level":null,"switch_type":null,"precheck_status":false,"t
ask_time":0,"fail_halt":false,"fail_upgrade":null,"component_names":null,"upgrade_t
o":null,"image_location":"file:///nfs/shared_storage/pca_firmware/X9-2/.../
ILOM-5_0_2_21_r140740-ORACLE_SERVER_X9-2-
rom.pkg","epld_image_location":null,"expected_iso_checksum":null,"checksum":null,"c
omposition_id":null,"request_id":"UWS-732d6fce-9f06-4329-b972-
d093bee40010","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 166
  Tasks 1 - Name = Validate Image Location
  Tasks 1 - Description = Verify that the image exists at the specified location
and is correctly named
  Tasks 1 - Time = 2021-09-24T11:18:32
[...]
```

At the end of the upgrade, the ILOM itself is rebooted automatically. However, the server component also needs to be rebooted for all changes to take effect. Wait 5 minutes to allow the ILOM workflow to complete first.

For minimum operational impact, schedule the compute node and management node reboot operations after all ILOMs have been patched. Take into account that rebooting the compute nodes requires migrating the compute instances. For more information, refer to "Performing Compute Node Operations" in the Hardware Administration chapter of the Oracle Private Cloud Appliance Administrator Guide.

> **Caution:**
>
> Always verify the cluster state before rebooting a management node. Consult the Oracle Private Cloud Appliance Release Notes for more information: refer to the known issue "Rebooting a Management Node while the Cluster State is Unhealthy Causes Platform Integrity Issues".

Upgrading the ZFS Storage Appliance Operating Software

To upgrade the operating software of the appliance's ZFS Storage Appliance, you enter the upgrade command with no additional parameters. The IP addresses of the storage controllers and the location of the firmware package are known, and a single upgrade command initiates a rolling upgrade of both controllers. If a new ILOM firmware version is included for the two controllers, it will be installed as part of the ZFS Storage Appliance upgrade process.

> **Caution:**
>
> Ensure that no users are not logged in to the ZFS Storage Appliance or the storage controller ILOMs during the upgrade process.
>
> Do not make storage configuration changes while an upgrade is in progress. While controllers are running different software versions, configuration changes made to one controller are not propagated to its peer controller.
>
> During firmware upgrade the storage controllers are placed in active/passive mode. They automatically return to active/active after the upgrade is completed.

Before You Begin

Before you initiate a ZFS Storage Appliance upgrade, you must disable the node state service to prevent errors in node states after the upgrade.

1. From a management node, set the provisioning lock by issuing this command:

   ```
   pca-admin locks set system provisioning
   ```

2. Perform the ZFS Storage Appliance upgrade using either the Service Web UI or the Service CLI procedure below.
3. Release the provisioning lock.

   ```
   pca-admin locks unset system provisioning
   ```

4. Confirm the lock state.

   ```
   pca-admin locks show system
   ```

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

The Create Request window appears. Choose *Upgrade* as the Request Type.

3. Select the appropriate upgrade request type: Upgrade Zfssa.
4. Fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **Image Location:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.

Using the Service CLI

1. Enter the upgrade command.

```
PCA-ADMIN> upgradeZfssa
Command: upgradeZfssa
Status: Success
Time: 2021-09-27 11:15:07,453 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1632914107346-zfssa-83002
Upgrade Request Id = UWS-881af57f-5dfb-4c75-8026-9f00cf3eb7c9
```

2. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id                              upgradeRequestId
commandName   result
  --                              ----------------
-----------   ------
  1632914107346-zfssa-83002       UWS-881af57f-5dfb-4c75-8026-9f00cf3eb7c9
zfssa         Passed
  1632926926773-host-32993        UWS-fef3b663-45b7-4177-a041-26f73e68848d
host          Passed
  1632990827394-host-56156        UWS-1a97a8d9-54ef-478d-a0c0-348a17ba6755
host          Passed
  1632990493570-host-6646         UWS-4c78f3ef-ac42-4f32-9483-bb43a309faa3
host          Passed

PCA-ADMIN> getUpgradeJob upgradeJobId=1632914107346-zfssa-83002
Command: getUpgradeJob upgradeJobId=1632914107346-zfssa-83002
Status: Success
Time: 2021-09-27 11:42:10,729 UTC
Data:
  Upgrade Request Id = UWS-881af57f-5dfb-4c75-8026-9f00cf3eb7c9
  Name = zfssa
  Start Time = 2021-09-29T11:15:07
  End Time = 2021-09-29T11:26:42
  Pid = 83002
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_zfssa_ak_2021_09_29-11.15.07.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":null,"result_ove
rride":null,"log_level":null,"switch_type":null,"precheck_status":false,"task_time"
:0,"fail_halt":false,"fail_upgrade":null,"component_names":null,"upgrade_to":null,"
image_location":"file:///nfs/shared_storage/pca_firmware/zfs/ak-
```

```
nas-2021.08.27-1.0x-
nondebug.pkg","epld_image_location":null,"expected_iso_checksum":null,"checks
um":null,"composition_id":null,"request_id":"UWS-881af57f-5dfb-4c75-8026-9f00
cf3eb7c9","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 697
  Tasks 1 - Name = Validate Image Location
  Tasks 1 - Description = Verify that the image exists at the specified
location and is correctly named
  Tasks 1 - Time = 2021-09-29T11:15:08
[...]
```

Upgrading the Switch Software

The appliance rack contains three categories of Cisco Nexus switches: a management switch, two leaf switches, and two spine switches. They all run the same Cisco NX-OS network operating software. **You must perform the upgrades in this order: leaf switches first, then spine switches, and finally the management switch**. Only one command per switch category is required, meaning that the leaf switches and the spine switches are upgraded in pairs.

Some versions of the network operating software consist of two files: a binary file and an additional EPLD (electronic programmable logic device) image. Both are automatically retrieved from their designated location during the upgrade process, and applied in the correct order.

Using the Service Web UI

1. In the navigation menu, click Upgrade & Patching.
2. In the top-right corner of the Upgrade Jobs page, click Create Upgrade or Patch.

   The Create Request window appears. Choose *Upgrade* as the Request Type.
3. Select the appropriate upgrade request type: Upgrade Switch.
4. Fill out the upgrade request parameters:
   - **Advanced Options JSON:** Optionally, add a JSON string to provide additional command parameters.
   - **EPLD:** This parameter is deprecated.
   - **Image Location:** This parameter is deprecated.
   - **Log Level:** Optionally, select a specific log level for the upgrade log file. The default log level is "Information". For maximum detail, select "Debug".
   - **Switch Type:** Select the switch type you intend to upgrade. The preferred upgrade order is as follows: leaf switches first, then spine switches, and finally the management switch.
5. Click Create Request.

   The new upgrade request appears in the Upgrade Jobs table.
6. When the upgrade has completed successfully, but other switches in the system still need to be upgraded, repeat this procedure for any other type of switch that requires upgrading.

Using the Service CLI

1. Determine the type of switch to upgrade (spine, leaf, management).

2. Enter the upgrade command.

   Syntax (entered on a single line):

```
upgradeSwitch
switchType=[MGMT | SPINE | LEAF]
```

   Example:

```
PCA-ADMIN> upgradeSwitch switchType=LEAF
Command: upgradeSwitch switchType=LEAF
Status: Success
Time: 2021-09-24 14:16:54,704 UTC
Data:
  Service request has been submitted. Upgrade Job Id = 1630511206512-cisco-20299
Upgrade Request Id = UWS-44688fe5-b4f8-407f-a1b5-8cd1b685c2c3
```

3. Use the request ID and the job ID to check the status of the upgrade process.

```
PCA-ADMIN> getUpgradeJobs
  id                              upgradeRequestId
commandName   result
  --                              ----------------
-----------   ------
  1632914107346-zfssa-83002       UWS-881af57f-5dfb-4c75-8026-9f00cf3eb7c9
zfssa         Passed
  1630511206512-cisco-20299       UWS-44688fe5-b4f8-407f-a1b5-8cd1b685c2c3
cisco         Passed
  1620921089806-ilom-21480        UWS-732d6fce-9f06-4329-b972-d093bee40010
ilom          Passed

PCA-ADMIN> getupgradeJob upgradeJobId=1630511206512-cisco-20299
Command: getupgradeJob upgradeJobId=1630511206512-cisco-20299
Status: Success
Time: 2021-09-24 15:48:08,455 UTC
Data:
  Upgrade Request Id = UWS-44688fe5-b4f8-407f-a1b5-8cd1b685c2c3
  Name = cisco
  Start Time = 2021-09-24T14:46:46
  End Time = 2021-09-24T14:59:44
  Pid = 20299
  Host = pcamn02
  Log File = /nfs/shared_storage/pca_upgrader/log/pca-
upgrader_cisco_firmware_2021_09_24-14.46.46.log
  Arguments =
{"verify_only":false,"upgrade":false,"diagnostics":false,"host_ip":null,"result_ove
rride":null,"log_level":null,"switch_type":"LEAF","precheck_status":false,"task_tim
e":0,"fail_halt":false,"fail_upgrade":null,"component_names":null,"upgrade_to":null
,"image_location":"file:///nfs/shared_storage/pca_firmware/network/cisco/
nxos.9.3.2.bin","epld_image_location":null,"expected_iso_checksum":null,"checksum":
null,"composition_id":null,"request_id":"UWS-44688fe5-b4f8-407f-
a1b5-8cd1b685c2c3","display_task_plan":false,"dry_run_tasks":false}
  Status = Passed
  Execution Time(sec) = 777
  Tasks 1 - Name = Validate Image Location
  Tasks 1 - Description = Verify that the image exists at the specified location
and is correctly named
  Tasks 1 - Time = 2021-09-24T14:46:47
[...]
```